TABLE OF CONTENTS

TABLE OF FIGURES

1.0 Introduction

The METS Importer is designed to import valid XML files into the MDB4.0 database.

In addition, there are two append functions that allow metadata to be appended to existing objects in the database.

A template instance is used to configure the data entry from the file into the database. The template instance is an XML file that maps each element or attribute in the XML file to a corresponding field in the MDB4.0 database.

A 'valid XML' is defined as a well formed XML file that validates against designated extension schemas. All files that are to be either imported or appended must meet the requirement of being a valid XML file. For this program, the METS Primary schema is the base schema with several optional extension schemas.

IMPORTANT NOTE:

The names of the files must follow the convention mets_object_id.xml, e.g. 'loc.mbrsrs.ca0001.ryb3441.xml'.

2.0 Getting Started

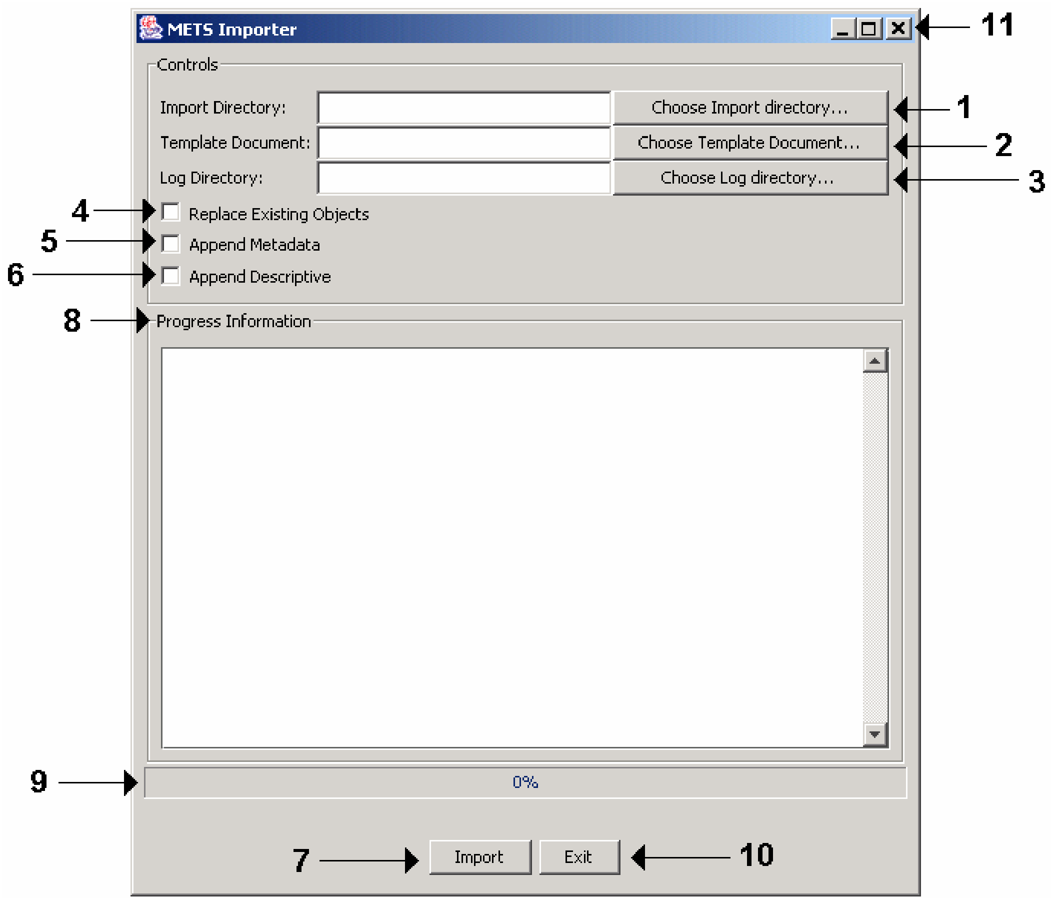


Figure 1 – User Interface

2.1 User Interface

<u>Figure 1</u>

1 – Browse to select a directory containing files to import.
2 – Browse to select a template file, e.g. mets_instance_v11.xml.
3 – Browse to select a directory to store the log files.
4 – Check to replace objects that already exist in the database.
5 – Check to append metadata to existing objects.
6 – Check to append descriptive metadata to an existing object.
7 – Click to run the program.
8 – Displays information to the user while the program is running.
9 – Indicates the percentage completed.
10 – Click to exit the program.
11 – Basic window functions from left to right: minimizes the window, maximizes the windows, and closes the window exiting the program.

2.2 Overview

There are three functions that Importer performs. They are Replace Existing Objects, Append Metadata, and Append Descriptive. The program only allows one option to be selected at a time. If an option is selected, and another option is checked, the program will unselect the first option. For importing entire objects, the only function that applies is Replace Existing Objects.

When a directory is chosen to import from, the program will check every file located in the directory to see that it conforms to the required standard. If a file does not conform, the file will be skipped and the program will continue.

2.3 Importing Objects: Step by Step

1. Start the program.
2. Choose the directory of files to import by typing the pathname in the box 'Import Directory' or clicking 'Choose Import Directory' (Figure 1 – 1).
3. Choose a template instance by typing the pathname in the box 'Template Document' or clicking 'Choose Template Document' (Figure 1 – 2). The template instance file should be in the same directory as the Importer program.
4. Choose a directory to store the generated log file by typing the pathname in the box 'Log Directory' or clicking 'Choose Log Directory' (Figure 1 – 3).
5. (Optional) Check box next to 'Replace Existing Objects' (Figure 1 – 4).
6. Run the program by clicking 'Import' (Figure 1 – 7).
7. 'Done – Import Completed.' will display in the Log (Figure 1 – 8) when the program has completed running.
8. Exit the program by clicking 'Exit' (Figure 1 – 10) or closing the window (Figure 1 – 11).

   Note: Once the program has finished running, repeat Step 2 and Step 6 to import files from another directory.

2.4 Explanation of Log Messages

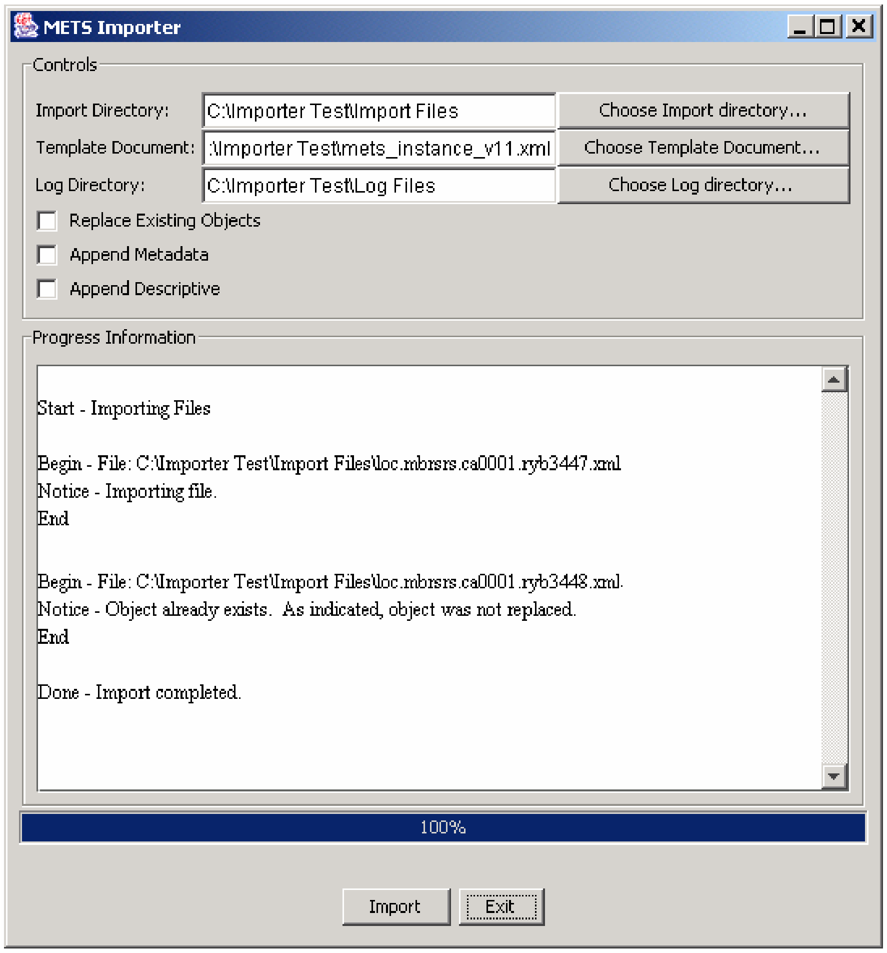
The log displays information about the files as they are being imported. This log is also automatically saved to a file named 'ImporterLog.txt' in the directory chosen in Step 4.

1. Start and Done: Indicate the beginning and end of the entire program run.

2. Begin and End: A 'Begin' and 'End' bracket each individual file. Information about the file displays between the 'Begin' and 'End' messages.

   Notice: A 'Notice' is information about the function being performed on the file. For example: "Notice – Importing file." means the designated file is in the process of being imported. Different 'Notice' messages will appear depending on the function selected while the program is running.

9. Error: An 'Error' message means there is a problem with the file and the object was not imported.



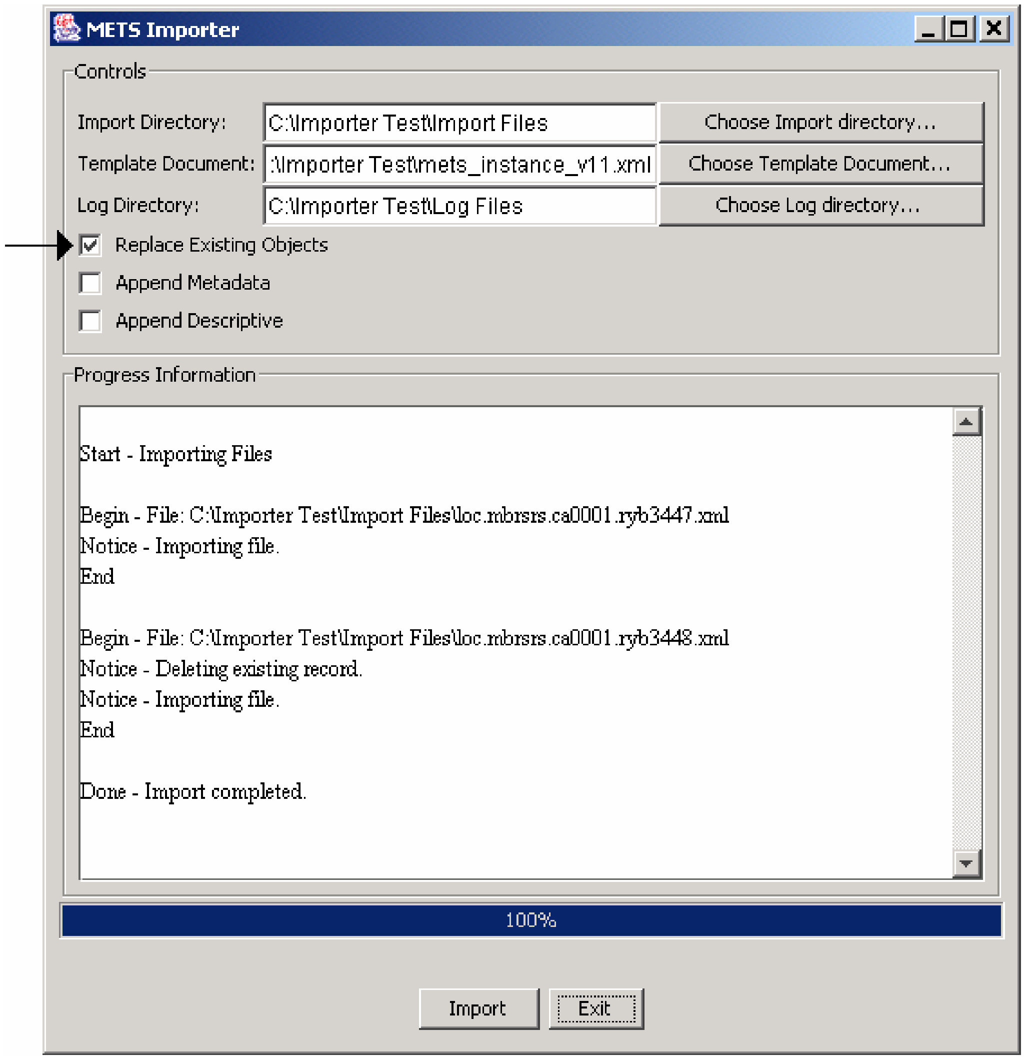
Import Directory: C:\Importer Test\Import Files
Template Document: :\Importer Test\mets_instance_v11.xml
Log Directory: C:\Importer Test\Log Files
☐ Replace Existing Objects
☐ Append Metadata
☐ Append Descriptive

```
Start - Importing Files

Begin - File: C:\Importer Test\Import Files\loc.mbrsrs.ca0001.ryb3447.xml
Notice - Importing file.
End

Begin - File: C:\Importer Test\Import Files\loc.mbrsrs.ca0001.ryb3448.xml.
Notice - Object already exists.  As indicated, object was not replaced.
End

Done - Import completed.
```

Figure 2 – User Interface after running program

3.0 Replace Existing Objects Function

The purpose of this function is to allow existing objects to be replaced. By default, if a file in the directory matches an existing object in the database, the program will skip over the file and no changes to the existing object will occur. Use this function to replace existing objects in the database. If a file in the directory matches an existing object, the existing object will be deleted and the file will be imported to create a new object. To select this function, check the box labeled 'Replace Existing Objects' (Figure 3).



Import Directory: C:\Importer Test\Import Files
Template Document: :\Importer Test\mets_instance_v11.xml
Log Directory: C:\Importer Test\Log Files

[✓] Replace Existing Objects
[ ] Append Metadata
[ ] Append Descriptive

```
Start - Importing Files

Begin - File: C:\Importer Test\Import Files\loc.mbrsrs.ca0001.ryb3447.xml
Notice - Importing file.
End

Begin - File: C:\Importer Test\Import Files\loc.mbrsrs.ca0001.ryb3448.xml
Notice - Deleting existing record.
Notice - Importing file.
End

Done - Import completed.
```

Figure 3 – User Interface with 'Replace Existing Objects' function

4.0 Append Functions

There are two functions provided that appends metadata to existing objects. They are the Append Metadata Function and the Append Descriptive Function.

4.1 Append Metadata

The purpose of this function is to add metadata to and update metadata in an existing object. The function allows metadata to be appended only at the top level of the structmap. At this time, the function will append the following types of metadata: rights, mdref, source, and descriptive. To select this function, check the box labeled ‘Append Metadata’ (Figure 4).

4.2 Append Descriptive Function

The purpose of this function is to append descriptive metadata to the top level of an existing object. To select this function, check the box labeled ‘Append Descriptive’ (Figure 5).

4.3 Appending Metadata: Step by Step

1. Start the program.

2. Choose the directory of files to import by typing the pathname in the box ‘Import Directory’ or clicking ‘Choose Import Directory’ (Figure 1 – 1).

3. Choose a template instance by typing the pathname in the box ‘Template Document’ or clicking ‘Choose Template Document’ (Figure 1 – 2). The template instance file should be in the same directory as the Importer program.

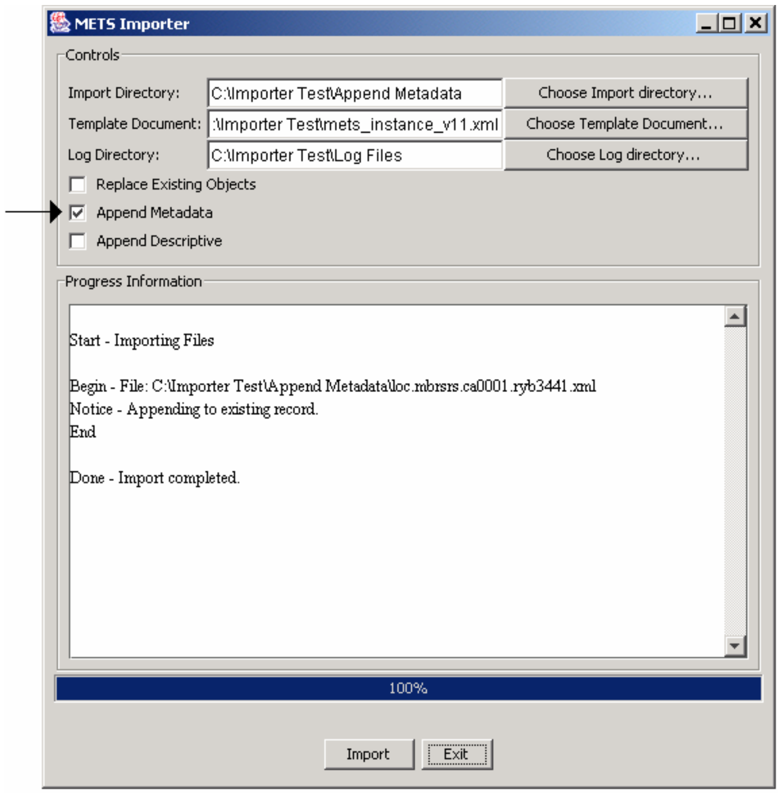
4. Choose a directory to store the generated log file by typing the pathname in the box ‘Log Directory’ or clicking ‘Choose Log Directory’ (Figure 1 – 3).

5. Check box next to ‘Append Metadata’ (Figure 4) or ‘Append Descriptive’ (Figure 5). The program only allows one option to be selected.

6. Run the program by clicking ‘Import’ (Figure 1 – 7).

7. ‘Done – Import Completed.’ will display in the Log (Figure 1 – 8) when the program has completed running.

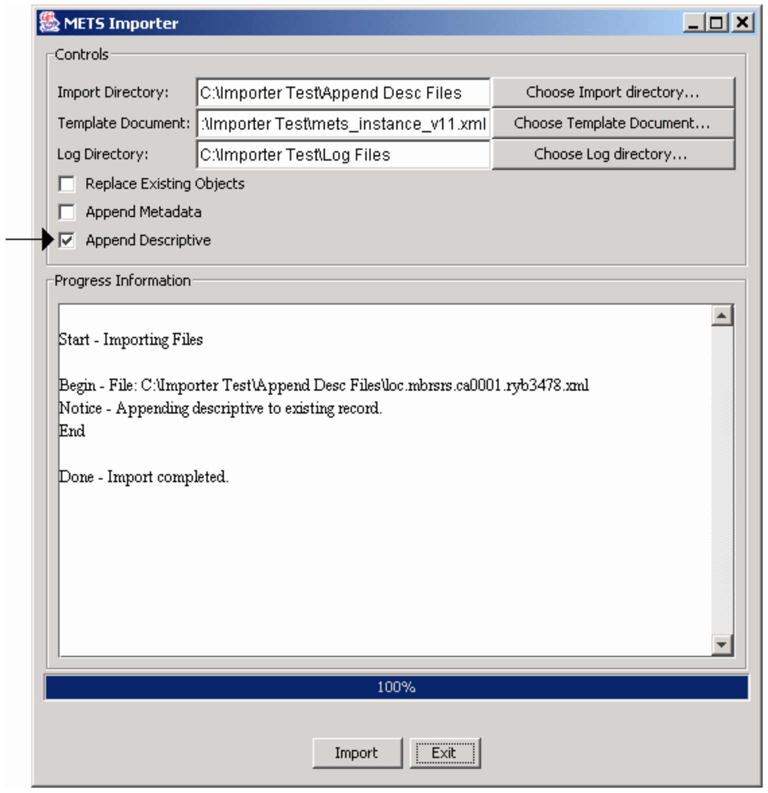
8. Exit the program by clicking ‘Exit’ (Figure 1 – 10) or closing the window (Figure 1 – 11).



Import Directory: C:\Importer Test\Append Metadata
Template Document: :\Importer Test\mets_instance_v11.xml
Log Directory: C:\Importer Test\Log Files
☐ Replace Existing Objects
☑ Append Metadata
☐ Append Descriptive

```
Start - Importing Files

Begin - File: C:\Importer Test\Append Metadata\loc.mbrsrs.ca0001.ryb3441.xml
Notice - Appending to existing record.
End

Done - Import completed.
```

Figure 4 – User Interface with 'Append Metadata' function



Import Directory: C:\Importer Test\Append Desc Files
Template Document: :\Importer Test\mets_instance_v11.xml
Log Directory: C:\Importer Test\Log Files
☐ Replace Existing Objects
☐ Append Metadata
☑ Append Descriptive

```
Start - Importing Files

Begin - File: C:\Importer Test\Append Desc Files\loc.mbrsrs.ca0001.ryb3478.xml
Notice - Appending descriptive to existing record.
End

Done - Import completed.
```

Figure 5 – User Interface with 'Append Descriptive' function